CAPÍTULO 5. Machine Learning

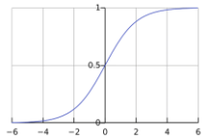
v.1.2 MARZO 2024

Ricardo Moraleda Gareta

[Director departamento de software de GDO Software]

Machine Learning

v.1.2 MARZO 2024



ML

R

RStudio

RStudio Cloud

7Days Machine Learning Challenge

Regresión logística

Transfor mación variables

Análisis exploratorio de datos

Análisis predictivo

Scoring

Data Science

MACHINE LEARNING

Machine Learning

Es un método de análisis de datos que automatiza la construcción de modelos analíticos. Es una rama de la inteligencia artificial basada en la idea de que los sistemas pueden aprender de datos, identificar patrones y tomar decisiones con mínima intervención humana.

La herramienta más utilizada para trabajar la modelización **predictiva** es a través del lenguaje R y de su IDE RStudio Desktop.

https://www.r-project.org/

https://www.rstudio.com/products/rstudio/download/

Para los ejemplos he utilizado IDE RStudio Cloud (web).

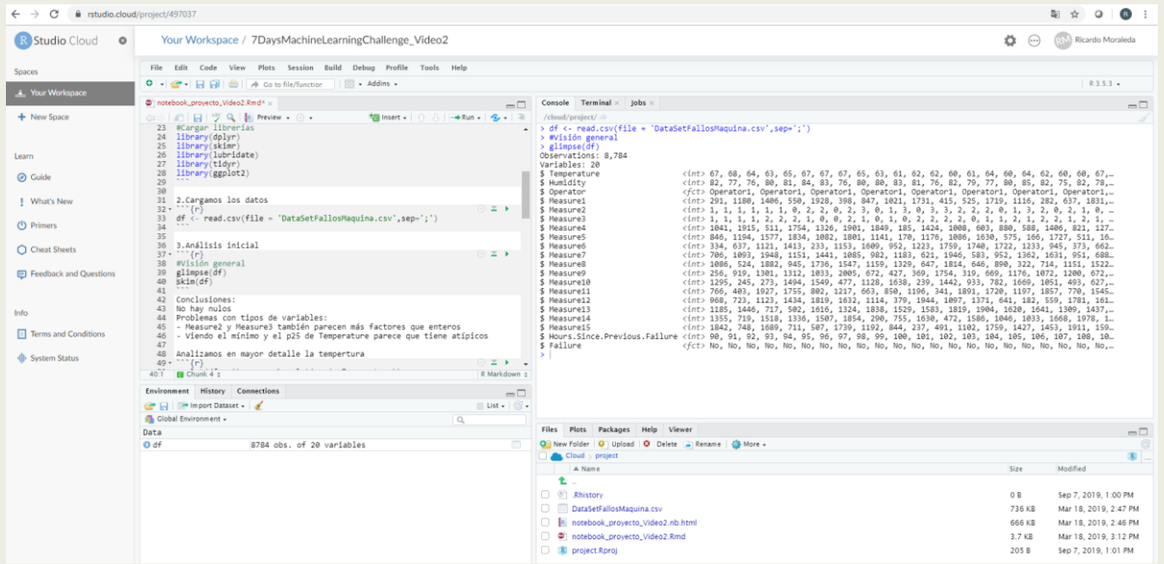
Si queréis utilizar Visual Studio .NET tenéis que bajaros estas tools. https://visualstudio.microsoft.com/es/vs/features/rtvs/

Extracto del curso Desafío Machine Learning de Isaac González.

https://datascience4business.com/

DS4B

El proyecto se basa en análisis predictivo a través de un input de datos (.CSV) de funcionamiento de una máquina para predecir posibles fallos (variable target Failure)



MACHINE LEARNING

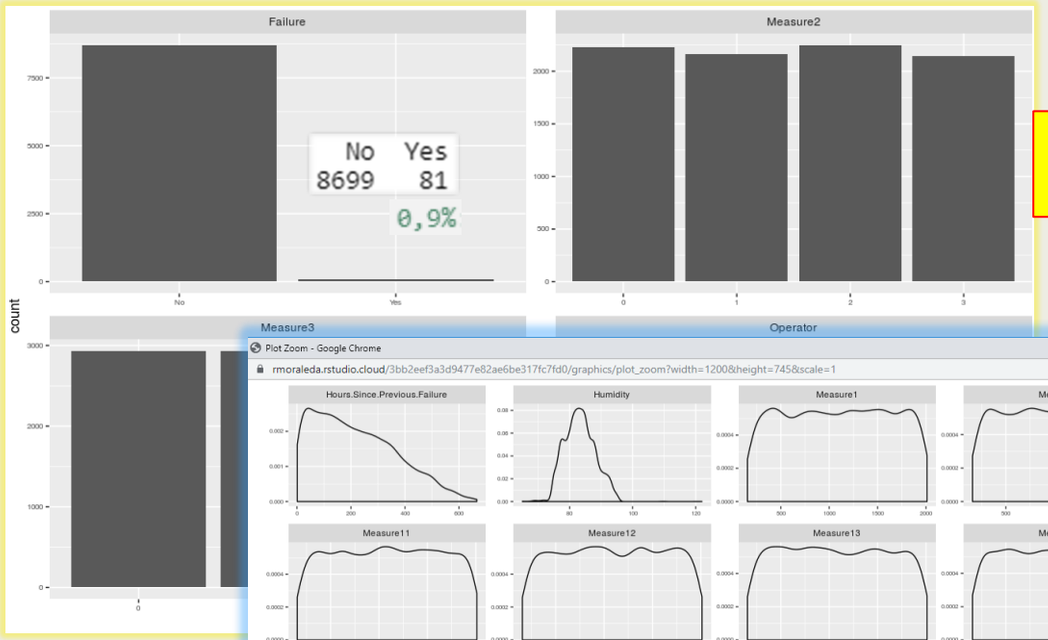
Machine Learning

Datos estadísticos con la función skim(data)

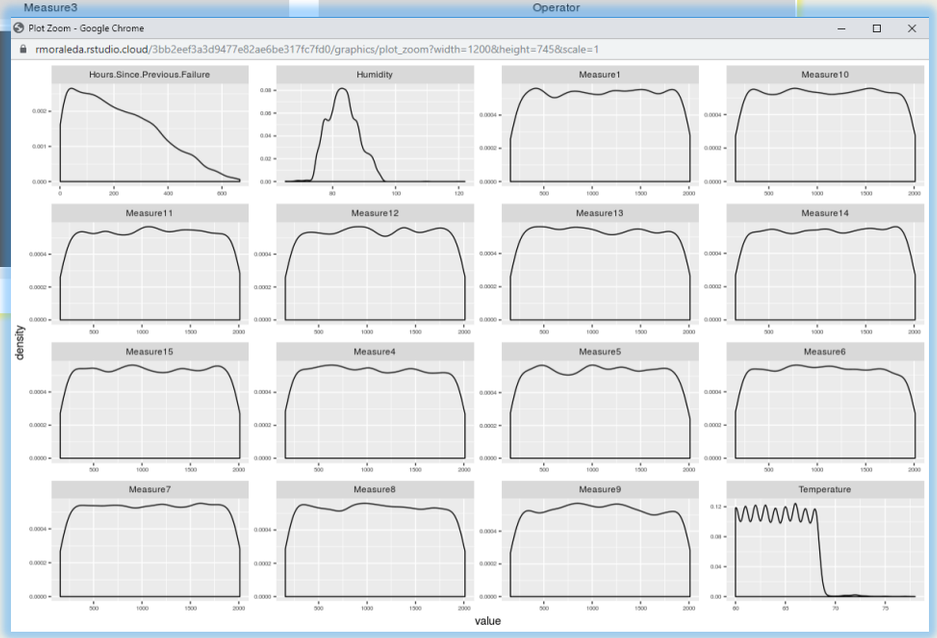
```
> skim(df)
Skim summary statistics
 n obs: 8780
 n variables: 20

── Variable type:factor ─────────────────────────────────────────────────────────────────────────────
 variable missing complete    n n_unique                       top_counts ordered
  Failure       0     8780 8780        2                  No: 8699, Yes: 81, NA: 0   FALSE
 Measure2       0     8780 8780        4      2: 2247, 0: 2227, 1: 2166, 3: 2140   FALSE
 Measure3       0     8780 8780        3        0: 2931, 1: 2928, 2: 2921, NA: 0   FALSE
 Operator       0     8780 8780        8 Ope: 1950, Ope: 976, Ope: 976, Ope: 976   FALSE

── Variable type:integer ────────────────────────────────────────────────────────────────────────────
                 variable missing complete    n    mean     sd  p0    p25    p50     p75 p100     hist
 Hours.Since.Previous.Failure       0     8780 8780  217.32 151.78   1  90     195    324     666 ▇▆▃▂▁▁▁▁
                 Humidity       0     8780 8780   83.34   4.84  65  80      83     87     122 ▁▂▇▃▁▁▁▁
                 Measure1       0     8780 8780 1090.82 537.04 155 629    1095.5 1555    2011 ▇▇▇▇▇▇▇▇
                Measure10       0     8780 8780 1082.37 537.59 155 619    1080   1547    2011 ▇▇▇▇▇▇▇▇
                Measure11       0     8780 8780 1088.55 534.84 155 627    1092.5 1549.25 2011 ▇▇▇▇▇▇▇▇
                Measure12       0     8780 8780 1088.23 533.23 155 627    1082   1552    2011 ▇▇▇▇▇▇▇▇
                Measure13       0     8780 8780 1076.79 535.08 155 609    1067.5 1539    2011 ▇▇▇▇▇▇▇▇
                Measure14       0     8780 8780 1088.46 537.28 155 617    1088.5 1560.25 2011 ▇▇▇▇▇▇▇▇
                Measure15       0     8780 8780 1082.4  537.63 155 614    1076   1550    2011 ▇▇▇▇▇▇▇▇
                 Measure4       0     8780 8780 1071.58 536.6  155 607.75 1058   1533    2011 ▇▇▇▇▇▇▇▇
                 Measure5       0     8780 8780 1075.77 533.19 155 606    1077   1541    2011 ▇▇▇▇▇▇▇▇
                 Measure6       0     8780 8780 1076.1  534.03 155 623    1072   1537    2011 ▇▇▇▇▇▇▇▇
                 Measure7       0     8780 8780 1087.09 538.16 155 621    1089   1558    2011 ▇▇▇▇▇▇▇▇
                 Measure8       0     8780 8780 1077.08 537.19 155 612    1073   1540.25 2011 ▇▇▇▇▇▇▇▇
                 Measure9       0     8780 8780 1082.14 532.98 155 631    1078   1532    2011 ▇▇▇▇▇▇▇▇
              Temperature       0     8780 8780   64.05   2.67  60  62      64     66      78 ▇▂▂▁▁▁▁▁
>
```

Video muy interesante de las librerías más comunes y útiles en R:
https://youtu.be/hY34j6_fvvg

Análisis exploratorio de datos (EDA) - plots



Variables factor



Variables integer

MACHINE LEARNING

Machine Learning

Previo a la construcción de un **4. modelo predictivo** y su **5. evaluación**, realizaremos → **1. muestreo**, **2. calidad de datos** y **3. preparación de variables** [como metodología recomendada]:

- Revisión de la calidad de los datos
- Eliminación de valores atípicos
- Es posible que se tenga que hacer transformación de variables
- Se balancea la variable target "Failure" para conseguir al menos un 10% de penetración del Sí (Yes) utilizando inframuestreo, antes era casi 1%.

Modelización

Predicción de la variable target (**Failure**) en base al resto

```
> target
[1] "Failure"
> indep
 [1] "Temperature"                 "Humidity"                    "Operator"
 [4] "Measure1"                    "Measure2"                    "Measure3"
 [7] "Measure4"                    "Measure5"                    "Measure6"
[10] "Measure7"                    "Measure8"                    "Measure9"
[13] "Measure10"                   "Measure11"                   "Measure12"
[16] "Measure13"                   "Measure14"                   "Measure15"
[19] "Hours.Since.Previous.Failure"
> formula <- reformulate(indep,target)
> formula
Failure ~ Temperature + Humidity + Operator + Measure1 + Measure2 +
    Measure3 + Measure4 + Measure5 + Measure6 + Measure7 + Measure8 +
    Measure9 + Measure10 + Measure11 + Measure12 + Measure13 +
    Measure14 + Measure15 + Hours.Since.Previous.Failure
>
```

Modelización con **REGRESIÓN LOGÍSTICA** para predecir el resultado de una variable dicotómica o binaria o booleana: **Failure**. Adopta 2 valores: 0 ó 1, es decir, fallo o no fallo.

```
> rl <- glm(formula,df_red,family=binomial(link='logit'))
> summary(rl) #Vemos el resultado

Call:
glm(formula = formula, family = binomial(link = "logit"), data = df_red)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.1509  -0.2149  -0.0992  -0.0395   3.6879
```

MACHINE LEARNING

Modelización. Entrenamiento

Resultados de coeficientes. Variables significativas o relevantes para poder predecir el resultado (tienen * o ***), es decir, Temperatute, Humidity y Measure9 → variables predictoras.

```
Coefficients:
                              Estimate   Std. Error z value      Pr(>|z|)
(Intercept)                -10.65712263  8.26147942  -1.290        0.1971
Temperature                  0.51282608  0.08802145   5.826 0.00000000567 ***
Humidity                    -0.32974373  0.05780685  -5.704 0.00000001169 ***
OperatorOperator2           -0.91538394  0.72204540  -1.268        0.2049
OperatorOperator3           -0.67780183  0.76404506  -0.887        0.3750
OperatorOperator4           -0.60196294  0.82473872  -0.730        0.4655
OperatorOperator5           -0.79570221  0.93631772  -0.850        0.3954
OperatorOperator6           -1.03339169  0.85778823  -1.205        0.2283
OperatorOperator7           -0.50560252  0.77920714  -0.649        0.5164
OperatorOperator8           -0.42556375  0.93014819  -0.458        0.6473
Measure1                     0.00011293  0.00037733   0.299        0.7647
Measure21                   -0.95334203  0.64934832  -1.468        0.1421
Measure22                   -0.64951086  0.60019907  -1.082        0.2792
Measure23                   -0.95055354  0.58716682  -1.619        0.1055
Measure31                    0.35858856  0.53579610   0.669        0.5033
Measure32                    0.26195630  0.53436315   0.490        0.6240
Measure4                     0.00026171  0.00040717   0.643        0.5204
Measure5                     0.00013602  0.00039528   0.344        0.7308
Measure6                     0.00036137  0.00039167   0.923        0.3562
Measure7                     0.00013098  0.00040347   0.325        0.7455
Measure8                     0.00043807  0.00042325   1.035        0.3007
Measure9                    -0.00085653  0.00042811  -2.001        0.0454 *
Measure10                    0.00058168  0.00039101   1.488        0.1368
Measure11                   -0.00029480  0.00041619  -0.708        0.4787
Measure12                   -0.00002978  0.00041699  -0.071        0.9431
Measure13                    0.00019108  0.00038442   0.497        0.6192
Measure14                    0.00043486  0.00038772   1.122        0.2620
Measure15                    0.00035073  0.00040263   0.871        0.3837
Hours.Since.Previous.Failure -0.00239746  0.00138218  -1.735        0.0828 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 519.53  on 776  degrees of freedom
Residual deviance: 186.25  on 748  degrees of freedom
AIC: 244.25

Number of Fisher Scoring iterations: 7
```

COEFICIENTES

Modelización. Entrenamiento

Eliminamos el resto de variables y volvemos a rehacer el modelo.

```
> indep_fin
[1] "Temperature" "Humidity"    "Measure9"
> formula <- reformulate(indep_fin,target) #actualizamos la fórmula
> rl <- glm(formula,df_red,family=binomial(link='logit'))
> summary(rl) #Vemos el resultado

Call:
glm(formula = formula, family = binomial(link = "logit"), data = df_red)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.3052  -0.2494  -0.1206  -0.0511   3.4442

Coefficients:
             Estimate  Std. Error z value       Pr(>|z|)
(Intercept) -10.9131014   7.2533423  -1.505         0.1324
Temperature   0.5061013   0.0809042   6.256 0.000000000396 ***
Humidity     -0.3020266   0.0467537  -6.460 0.000000000105 ***
Measure9     -0.0008612   0.0003766  -2.287         0.0222 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 519.53  on 776  degrees of freedom
Residual deviance: 205.70  on 773  degrees of freedom
AIC: 213.7

Number of Fisher Scoring iterations: 7
```

COEFICIENTES

Aplicación del modelo a varios datos. Incremento de 10 º de la **temperatura** de 50º a 60º hace que la prob. de fallo ↑↑ del 24% al 98 % pero si la **humedad** sube al 70% la prob. de fallo ↓↓ al 11%.

Variable	*Coeficiente*	*Dato1*	*Dato2*	*Dato3*
(Intercept)	-10,9131014	1	1	1
Temperature	0,5061013	50	60	60
Humidity	-0,3020266	50	50	70
Measure9	-0,0008612	500	500	500
		-1,14	3,92	-2,12
	% PROB.	24,23	98,06	10,72

Finalmente aplicamos la predicción de nuestro modelo ya entrenado a los datos (scoring). Damos como fallo > 80% probabilidad predicción y > 60%.

```
> df$scoring <- predict(rl,df,type='response')
> #Como la penetración inicial era del 1%, vamos a poner un punto de corte muy alto, por ejemplo por encima del 80%
> df$prediccion <- ifelse(df$scoring > 0.8,1,0)
> table(df$prediccion)

   0    1
8728   52
>
```

```
> #Vamos a ver qué pasa si bajamos la decisión al 60%
> df$prediccion <- ifelse(df$scoring > 0.6,1,0)
> table(df$prediccion,df$Failure)

     No  Yes
  0 8687   24
  1   12   57
```

MACHINE LEARNING

Evaluación del modelo

Se comprueba el modelo o predicción con los datos reales de fallo en base al criterio del scoring del 80% y del 60%. (se utiliza como indicador la matriz de confusión)

80%

```
> table(df$prediccion,df$Failure)

     No  Yes
  0 8698   30
  1    1   51
>
```

60%

```
> table(df$prediccion,df$Failure)

     No  Yes
  0 8687   24
  1   12   57
>
```

Predicción

Realidad

Donde:
VP: Verdaderos Positivos
FP: Falsos Positivos
FN: Falsos Negativos
VN: Verdaderos Negativos

Cuando realidad y modelo (predicción) coinciden se considera un acierto (VP + VN = 8749) y si no un error (FP + FN = 31).

Esto significa que la predicción de no fallo (0) en la realidad hay 8698 muestras que no son fallos (VN) y 30 que sí (FN).

Esto significa que la predicción de fallo (1) en la realidad hay 51 muestras que sí son fallos (VP) y 1 que no (FP).

Según el valor de corte 80% o más permisivo (bajando a 60%) cambian los datos, o me dejo fallos y no los predigo o predigo fallos que no son. Ese criterio tiene una ciencia detrás.

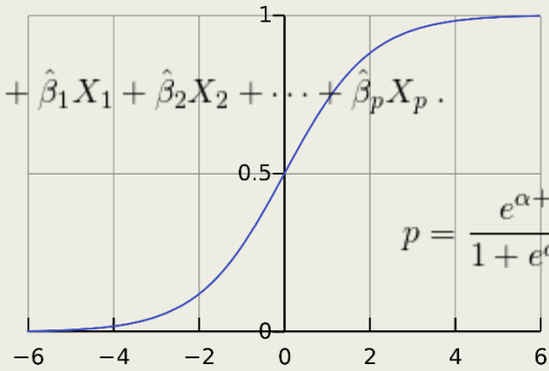
Además se pueden sacar los indicadores opuestos como son: Precisión → VP/ (VP+FP) = 98,07% y Cobertura →VP / (VP+FN) = 62,96%.

Recomiendo leer el libro "Big Data para CEOs y Directores de Marketing" de Isaac González para entender todos estos conceptos.

Resumen

- Caso del mundo industrial. Predicción de mantenimientos preventivos (antes del fallo).
- Modelizar y entrenar el modelo para sacar los coeficientes (pesos) de cada variable predictora para obtener el target.
- Poner en producción (C#, SCADA, etc.) aplicando en este caso las fórmulas de regresión logística, pasándolo a una probabilidad (p-%) y generando una alarma si supera el umbral predefinido o % de probabilidad de fallo.

$$\log\left(\frac{p}{1-p}\right) = \hat{\alpha} + \hat{\beta}_1 X_1 + \hat{\beta}_2 X_2 + \cdots + \hat{\beta}_p X_p \,.$$



$$p = \frac{e^{\alpha+\beta_1 X_1+\beta_2 X_2+\cdots+\beta_p X_p}}{1+e^{\alpha+\beta_1 X_1+\beta_2 X_2+\cdots+\beta_p X_p}}$$

https://es.wikipedia.org/wiki/Regresión_logística

DATA SCIENCE

Data Science

Video sobre Análisis con Data Science y R de lo que Google sabe sobre ti (Isaac González). Puedes ver todo el ejercicio y replicarlo con tus propios datos en https://youtu.be/33CuFx0j0zQ

A través de un fichero de entrada JSON del historial de ubicaciones de Google. De inicio 10.366 registros con 9 variables. Tras hacer varias transformaciones 14 variables.

```
> glimpse(df)
Observations: 10,366
Variables: 9
$ timestampMs      <chr> "1416381188595", "1416381264340", "1416381327324", "1416381388765", "1416381447…
$ latitudeE7       <int> 413956688, 413957502, 413951955, 413921003, 413920030, 413954096, 413954096, 41…
$ longitudeE7      <int> 21938919, 21939394, 21933333, 21975261, 22033423, 21935327, 21935327, 21935327,…
$ accuracy         <int> 27, 55, 29, 37, 58, 588, 588, 588, 588, 588, 4, 4, 588, 588, 588, 588, 588, 588…
$ activity         <list> [<data.frame[1 x 2]>, NULL, <data.frame[1 x 2]>, NULL, NULL, <data.frame[1 x 2…
$ velocity         <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,…
$ heading          <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,…
$ altitude         <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,…
$ verticalAccuracy <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,…
>
```

```
> glimpse(df)
Observations: 10,366
Variables: 14
$ accuracy         <int> 27, 55, 29, 37, 58, 588, 588, 588, 588, 4, 4, 588, 588, 588, 588, 588, 588, 588…
$ velocity         <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, …
$ heading          <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, …
$ altitude         <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, …
$ verticalAccuracy <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, …
$ time             <dttm> 2014-11-19 07:13:08, 2014-11-19 07:14:24, 2014-11-19 07:15:27, 2014-11-19 07:16:28, 2014-11-19 07:17:27, 20…
$ fecha            <date> 2014-11-19, 2014-11-19, 2014-11-19, 2014-11-19, 2014-11-19, 2014-11-19, 2014-11-19, 2014-11-19, 2014-11-19,…
$ hora             <chr> "7:13", "NA:NA", "7:15", "NA:NA", "NA:NA", "7:18", "NA:NA", "7:20", "NA:NA", "7:22", "NA:NA", "NA:NA", "7:25…
$ latitud          <dbl> 41.39567, 41.39575, 41.39520, 41.39210, 41.39200, 41.39541, 41.39541, 41.39541, 41.39541, 41.44420, 41.44944…
$ longitud         <dbl> 2.193892, 2.193939, 2.193333, 2.197526, 2.203342, 2.193533, 2.193533, 2.193533, 2.193533, 2.205538, 2.198340…
$ act.time         <dttm> 2014-11-19 07:13:17, NA, 2014-11-19 07:15:30, NA, NA, 2014-11-19 07:18:46, NA, 2014-11-19 07:20:45, NA, 201…
$ act.fecha        <date> 2014-11-19, NA, 2014-11-19, NA, NA, 2014-11-19, NA, 2014-11-19, NA, 2014-11-19, NA, NA, 2014-11-19, NA, NA,…
$ act.dia.semana   <ord> Wednesday, NA, Wednesday, NA, NA, Wednesday, NA, Wednesday, NA, Wednesday, NA, NA, Wednesday, NA, NA, Wednes…
$ act.actividad    <fct> STILL, NULL, STILL, NULL, NULL, IN_VEHICLE, NULL, IN_VEHICLE, NULL, IN_VEHICLE, NULL, NULL, IN_VEHICLE, NULL…
```

Análisis de datos de Discovery y generación de insights

Rango de fechas de las medidas

```
> summary(df$fecha)
        Min.      1st Qu.       Median         Mean      3rd Qu.         Max.
"2014-11-19" "2014-12-10" "2014-12-14" "2015-02-05" "2014-12-17" "2018-07-21"
```

Cantidad de muestras de media por día y por hora (muestras cada 3 min.)

```
> df %>% group_by(fecha) %>% summarise(conteo = n()) %>% summarise(media = mean(conteo))
# A tibble: 1 x 1
  media
  <dbl>
1  432.
> df %>% group_by(fecha) %>% summarise(conteo = n()) %>% summarise(media = mean(conteo)) / 24
     media
1 17.99653
```

Precisión posición. Mediana error es 37 m y el 75% de las medidas < 51 m.

```
> summary(df$accuracy)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    3.0    25.0    37.0   202.3    51.0  3177.0
```

Distintas actividades realizadas (según algoritmo de Google)

```
> distinct(df, act.actividad)
    act.actividad
1           STILL
2            NULL
3      IN_VEHICLE
4         ON_FOOT
5         UNKNOWN
6         TILTING
7 EXITING_VEHICLE
8      ON_BICYCLE
```

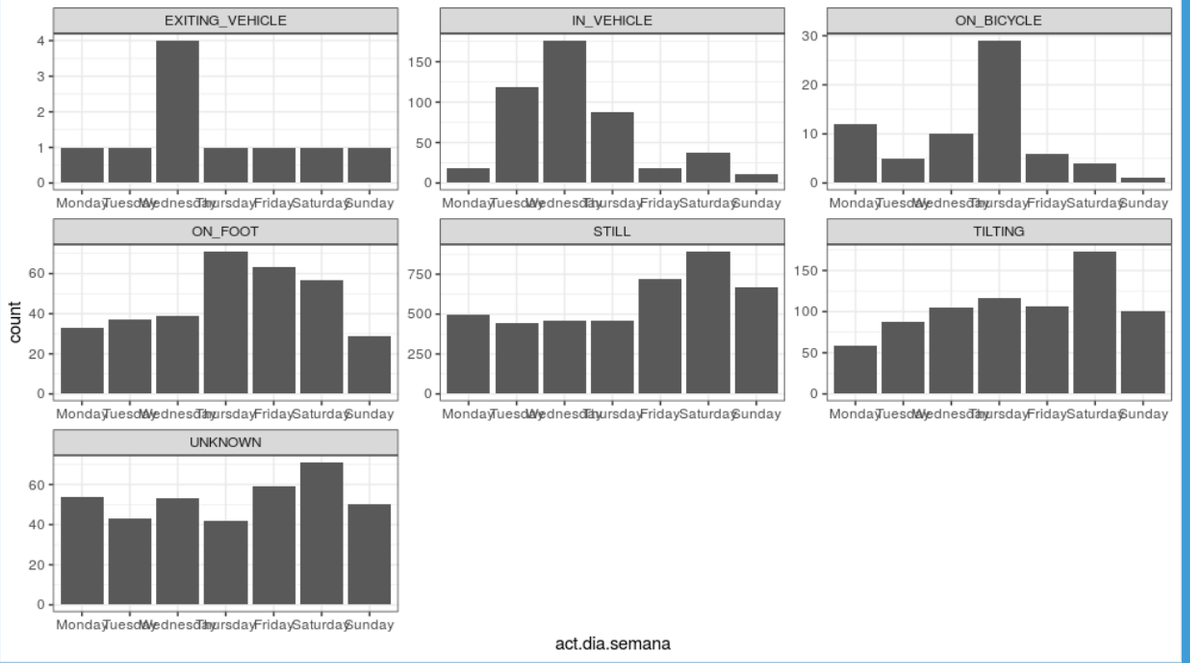
```
> count(df, act.actividad, sort = T)
# A tibble: 8 x 2
  act.actividad       n
  <fct>           <int>
1 NULL             4236
2 STILL            4136
3 TILTING           749
4 IN_VEHICLE        467
5 UNKNOWN           372
6 ON_FOOT           329
7 ON_BICYCLE         67
8 EXITING_VEHICLE    10
```

DATA SCIENCE

Análisis de datos de Discovery y generación de insights

Actividades por día de la semana:

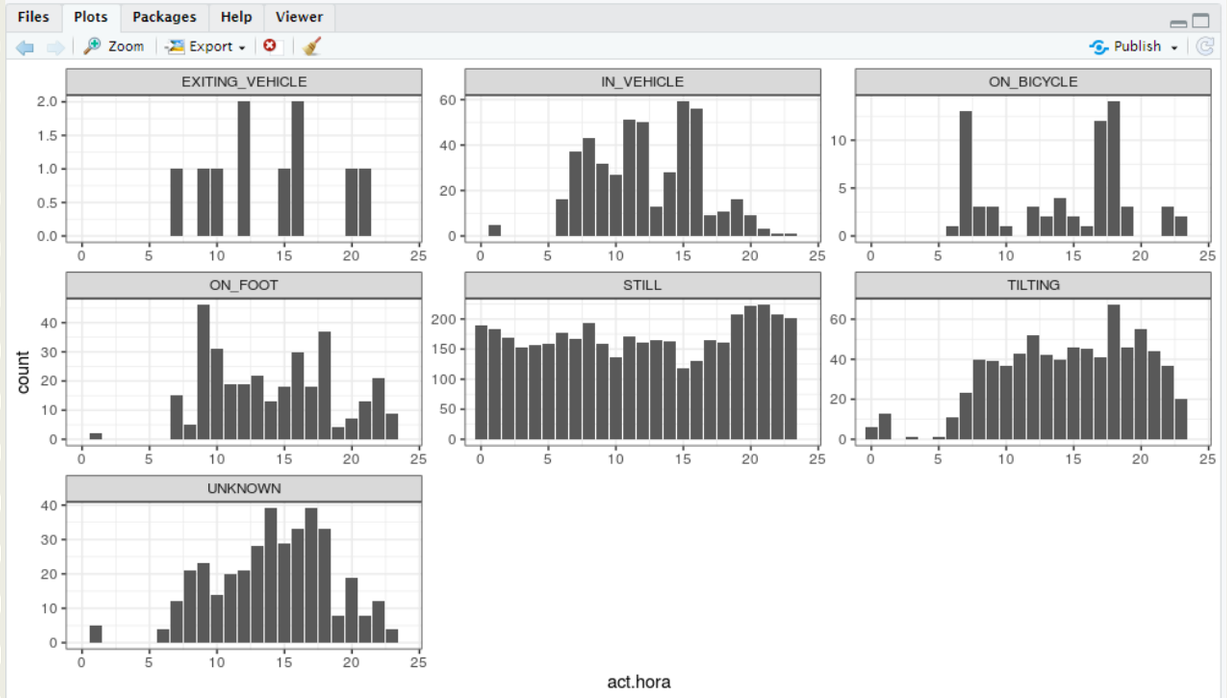
```
> df %>%
+   filter((!is.na(act.actividad)) & (!is.na(act.dia.semana))) %>%
+   ggplot(aes(act.dia.semana)) +
+   geom_bar() +
+   facet_wrap(~act.actividad, scales = 'free') + theme_bw()
>
```



Análisis de datos de Discovery y generación de insights

Actividades por hora del día:

```
> df %>%
+   filter((!is.na(act.actividad)) & (!is.na(act.hora))) %>%
+   ggplot(aes(act.hora)) +
+   coord_cartesian(xlim = c(0,24)) +
+   geom_bar() +
+   facet_wrap(~act.actividad, scales = 'free') + theme_bw()
>
```

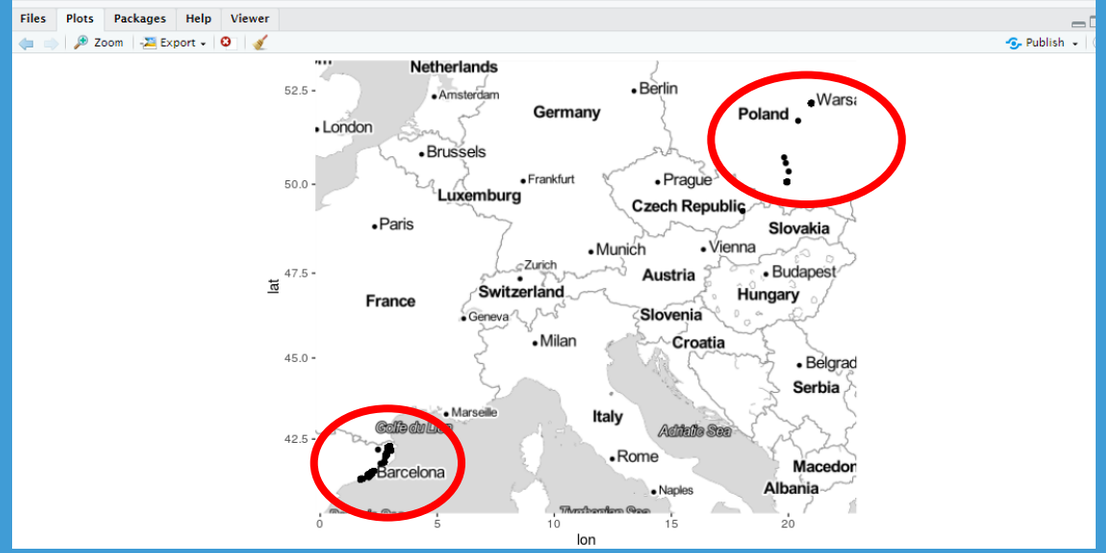


DATA SCIENCE

Análisis de datos de Discovery y generación de insights

Geolocalización de los puntos en un mapa. Existen otras maneras de representación con los mapas de Google a través de su API.

```
> altura <- max(df$latitud) - min(df$latitud)
> anchura <- max(df$longitud) - min(df$longitud)
>
> bordes <- c(bottom = min(df$latitud) - 0.1 * altura,
+             top = max(df$latitud) + 0.1 * altura,
+             left = min(df$longitud) - 0.1 * anchura,
+             right = max(df$longitud) + 0.1 * anchura)
>
> map <- get_stamenmap(bordes, zoom = 5, maptype = "toner-lite")
>
> ggmap(map, zoom = 5) + geom_point(data = df, mapping = aes(x=longitud, y = latitud))
>
```



Resumen

- Caso de uso sobre lo que Google sabe sobre mí.
- Horarios, actividades, localizaciones.
- Ejercicio analizado con la potente herramienta R.
- Con todos estos datos, Google puede cruzarlos con indicadores de puntos de interés para sacar conclusiones sobre tus gustos, aficiones, tiendas, publicidad, etc.
- Ejemplo muy bueno y muy interesante por el tipo de información personal analizada y todas las interpretaciones que uno puede hacerse así mismo.

Lecturas y recursos

Lecturas recomendadas

2017

Recursos

- R

https://www.r-project.org/

- Formación en ML / Data Scientist

https://datascience4business.com/

- Regresión logística

https://es.wikipedia.org/wiki/Regresi%C3%B3n_log%C3%ADstica

Machine Learning
v.1.2 MARZO 2024

https://www.linkedin.com/in/ricardo-moraleda-gareta-9421099

https://www.linkedin.com/company/gdo-electric1996/

RICARDO MORALEDA GARETA

GDO SOFTWARE